

Percolation Optique

Cécile BRÜTT, Maxence GRANDADAM, Victor PY
ESPCI ParisTech, Promotion 133

Mai 2016

1 Matériel

1. Pointeur LASER 405nm \pm 10nm + Support melles griot approprié
2. Alimentation stabilisée Métrix 413D
(équivalent plus actuel : <http://www.chauvin-arnoux.com/fr/produits/17/50>)
3. 2 miroirs P100A + pieds Thorlabs
4. Support élévateur Boy 105
<http://www.labomoderne.com/gamme.supports-elevateurs-boy-.BOY100.html>
5. Plaque de plexiglas; fabrication à l'atelier
6. 2 lamelles verre microscope VWR
7. Billes en hydrogel gonflable / Cylindres en PDMS
8. 2 prismes (forme et taille quelconques)
9. Lentille thorlabs convergente f=35mm
10. Photodétecteur Thorlabs PDA100A-EC
<http://www.thorlabs.com/thorProduct.cfm?partNumber=PDA100A-EC>
11. Support 2 axes Thorlabs
12. Plaque Thorlabs
13. Oscilloscope Agilent technologies DSO1052B

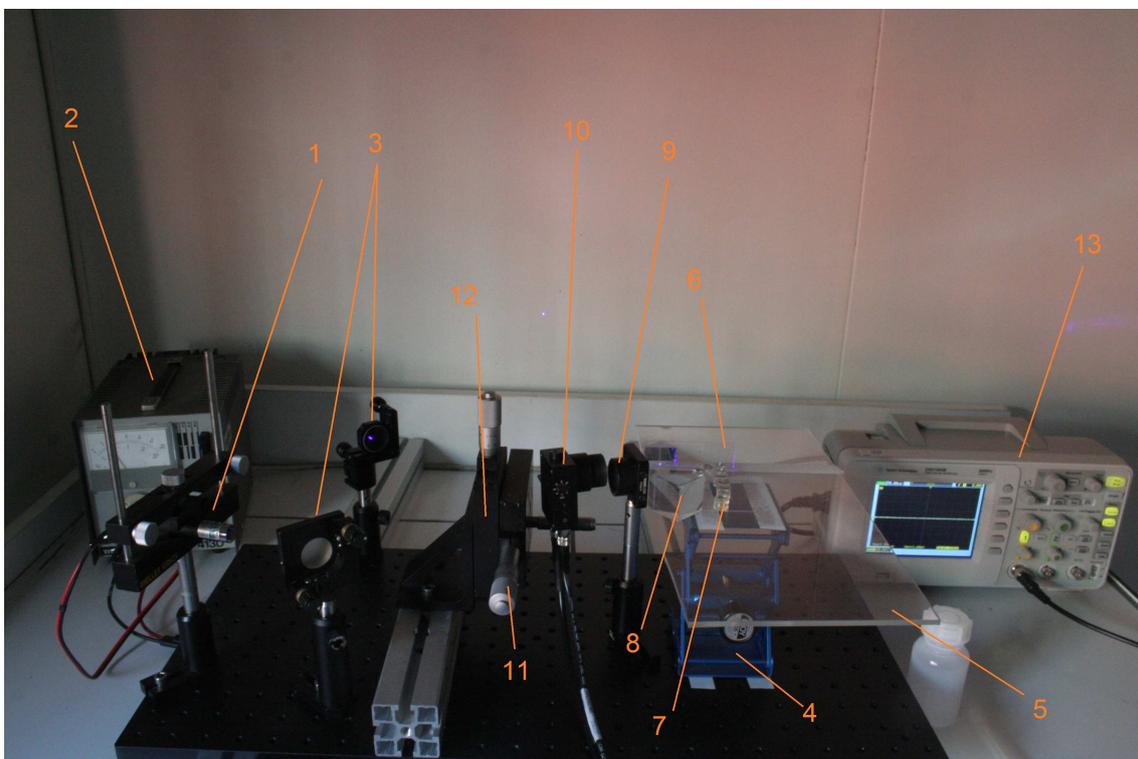


Figure 1: Légende du matériel utilisé pour le montage

2 Méthodes

2.1 LASER

Le LASER est alimenté par une alimentation stabilisée délivrant environ 5V, les piles se déchargeant trop vite. Il y est relié par des pinces crocodiles entre le ressort intérieur et la carcasse du LASER.

2.2 Miroirs et dispositif d'entrée

Les miroirs sont orientés de telle façon que le LASER entre dans le prisme avec un angle supérieur à l'angle critique de Snell-Descartes.

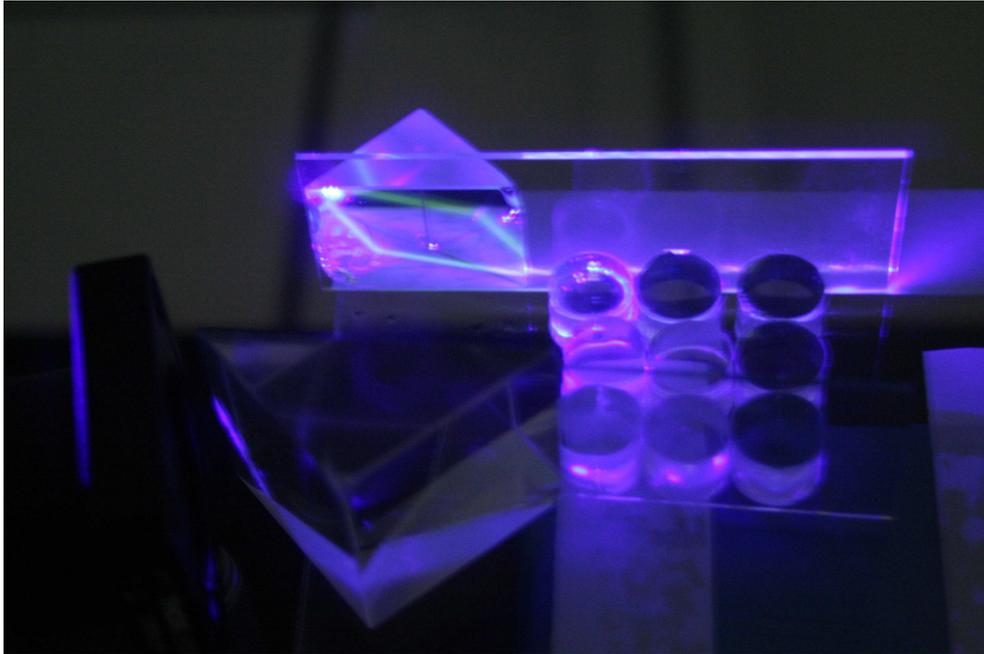


Figure 2: Photo montrant l'angle d'entrée de la lumière dans le prisme

La lame de verre permet de transmettre le rayon au bon angle à plus grande distance (environ 6cm) pour, par exemple, créer plusieurs chemins lumineux.

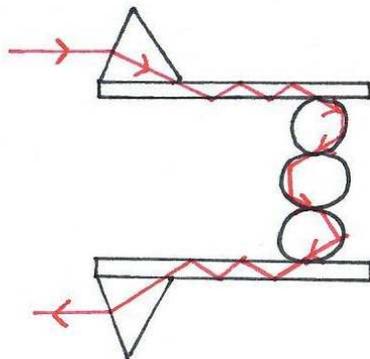


Figure 3: Traversée de la lumière depuis le prisme d'entrée

2.3 Chemin(s) de billes ou de cylindres

Les chemins de percolation sont réalisés avec des billes transparentes d'hydrogels de composition inconnue (billes utilisées pour retenir l'eau dans les plantes), ou en cylindres de PDMS fabriqués à l'IPGG avec Philippe Nghe. Les cylindres et les billes ont un diamètre de l'ordre du cm, les cylindres étant un petit peu plus petits que les billes. Les billes gonflant dans l'eau, elles ne sont pas monodisperses ce qui pose des problèmes dans la réalisation du montage : la lumière ne reste pas dans le plan du LASER. Les cylindres, eux, sont fabriqués grâce à un moule (plaque de plexiglas trouée au cutter LASER de l'atelier) et sont donc assez monodisperses. Par contre, on a pu remarquer qu'ils gonflent légèrement avec l'humidité de l'air. Leur flexibilité dépend de la durée passée à l'étuve pour le séchage (au moins 4h, pour certains tout un week-end). L'indice optique des billes est très proche de 1,33 (indice de l'eau dans laquelle elles ont gonflé), et celui des cylindres est de l'ordre de 1,4.

La lumière arrivant avec un angle supérieur à l'angle critique ne peut pas traverser l'interface hydrogel/air et est donc indéfiniment réfléchiée dans la bille. Cette réflexion n'est évidemment jamais parfaite. Lorsqu'on accole une seconde bille à la première, le milieu ne changeant pas d'indice, la lumière se propage en ligne droite d'une bille à l'autre. Mais cela représente le cas idéal : en effet, les contacts non plus ne sont jamais parfaits, si bien qu'une partie non négligeable de la lumière incidente est diffusée de façon supposément isotrope à partir du point de contact. Une certaine quantité peut alors pénétrer dans la bille suivante sous l'angle critique, mais le reste est a priori perdu. Afin de minimiser cet effet, on ajoute du pseudo-liquide d'indice (huile de silicone) avec une pipette pasteur sur chacun des contacts bille-bille et bille/plaque.

Le logiciel de simulation d'optique géométrique OptGeo permet d'obtenir simplement des simulations de traversée par la lumière d'un système prisme-bille avec et sans les réflexions parasites.

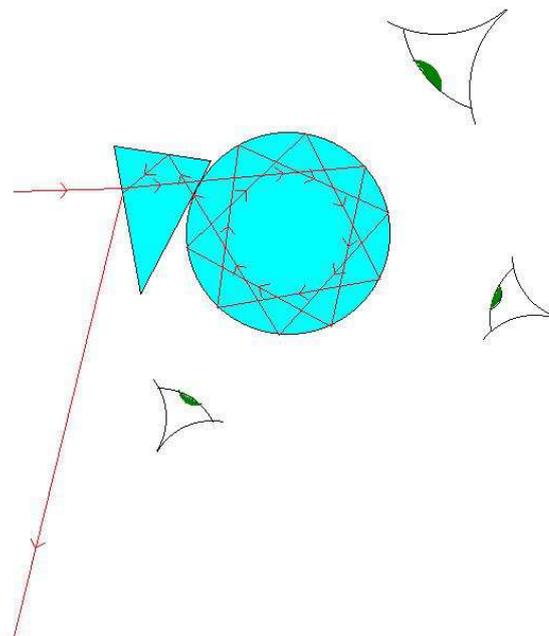


Figure 4: Simulation OptGeo : Traversée de la lumière dans une bille dans une situation "parfaite"

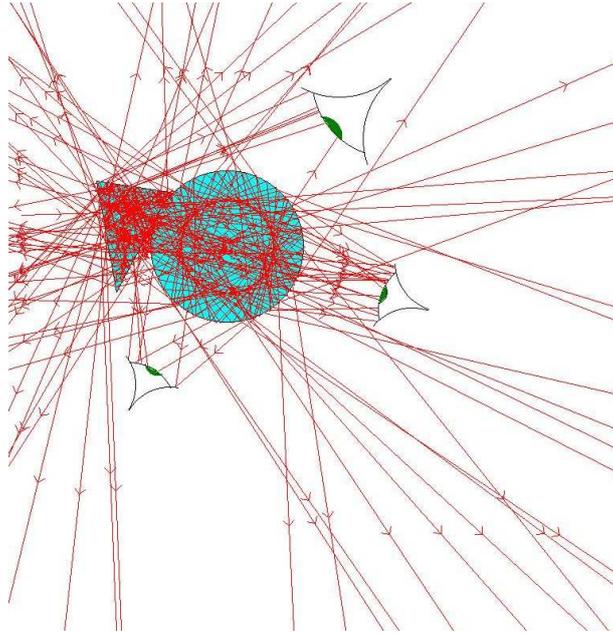


Figure 5: Simulation OptGeo : Traversée de la lumière dans une bille en prenant en compte les phénomènes de diffusion

Sur les simulations, on voit se dessiner le cercle lumineux qui délimite le faisceau des rayons qui se réfléchissent dans la bille ou le cylindre. On constate également l'extrême pollution lumineuse qui découle de l'imperfection des réflexions.

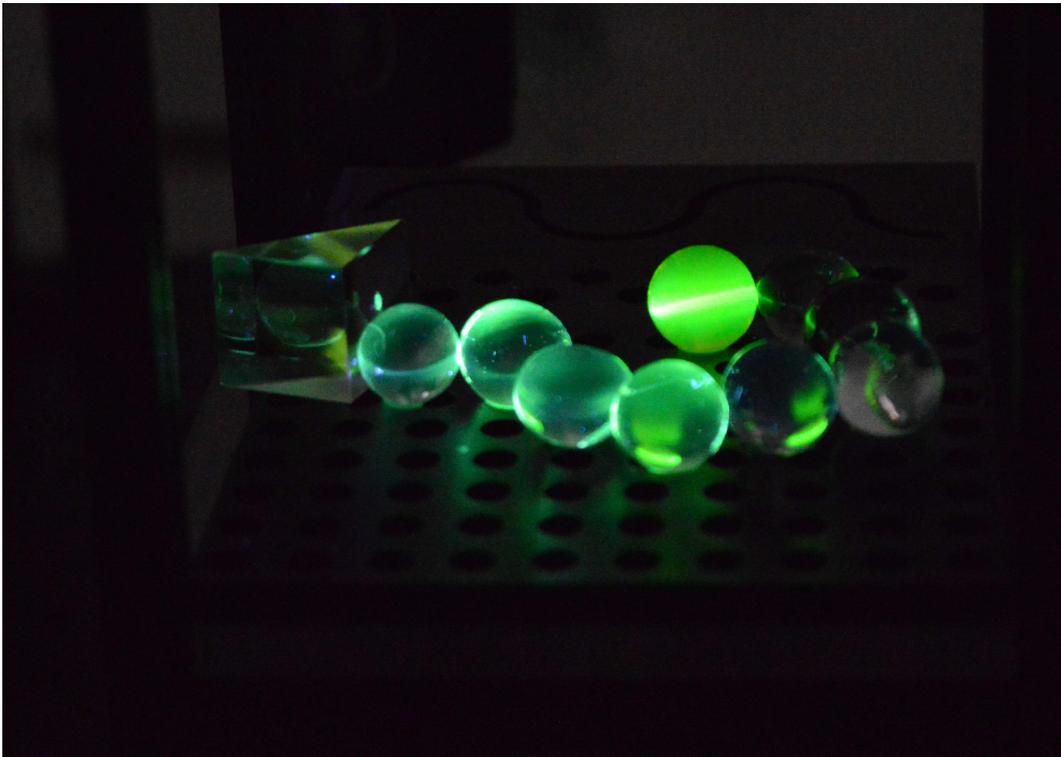


Figure 6: Visualisation du cercle caractéristique de la percolation

2.4 Prisme et lames de sortie, photorécepteur

La lumière en sortie des chemins de billes est récupérée par la lame de microscope puis par le prisme de sortie qui redirige le faisceau. Ce dernier est focalisé sur le photorécepteur grâce à une lentille convergente de focale très courte (35mm) pour que le photorécepteur reçoive *toute* la lumière issue du montage, et ainsi pouvoir faire une analyse quantitative de la lumière reçue. Le photorécepteur, branché sur l'oscilloscope, donne une tension proportionnelle à l'intensité lumineuse reçue. Son gain peut être ajusté pour obtenir un signal correct, mais ne doit pas être trop élevé pour ne pas saturer le capteur. En ordre de grandeur, le niveau de bruit (dans le noir, avec le LASER mais sans chemin de billes) que nous avons mesuré est de l'ordre de 20mV, et le signal pour un chemin d'une bille pouvait atteindre 3V (dans le noir également).

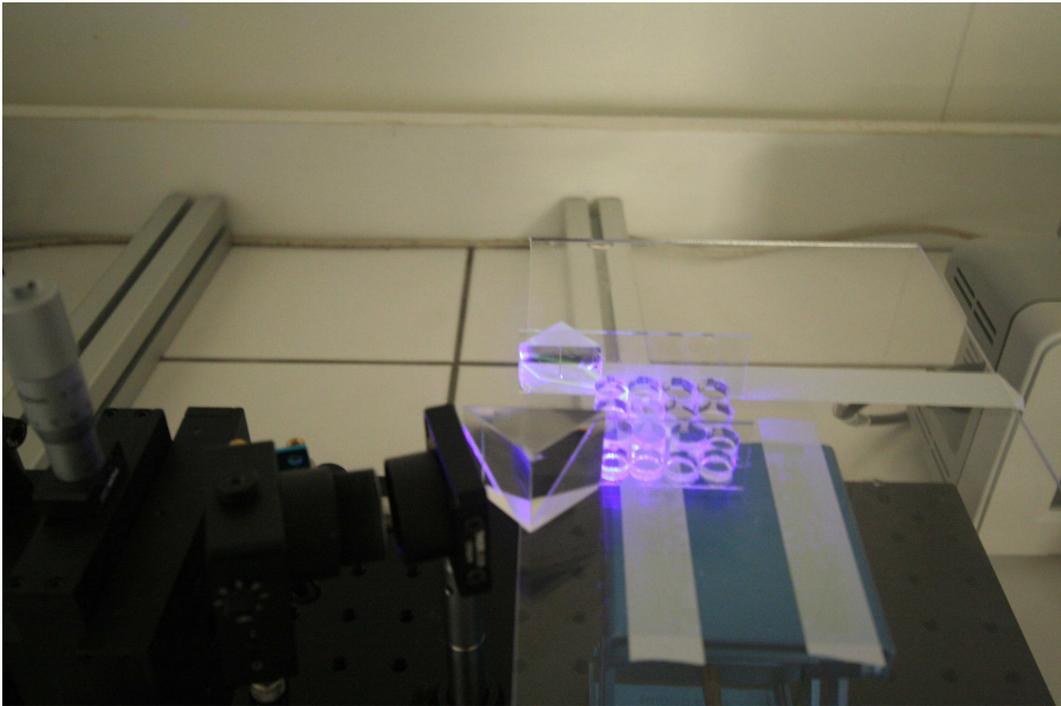


Figure 7: Photographie du montage au niveau des billes ou cylindres

3 Théorie du réseau

Notre but premier était de retrouver la valeur du seuil de percolation 2D d'un réseau aléatoire carré de taille N de cylindres transparents (probabilité d'occurrence P_{occ}) et de cylindres en métal obstrueteurs de chemins (probabilité d'occurrence $1 - P_{occ}$). Le seuil de percolation est défini comme la probabilité d'occurrence P_{occ} à partir de laquelle la probabilité de percolation P_{perc} (i.e. que la lumière se fraie un chemin de la première à la dernière boule) devient non nulle. Intuitivement toutefois, on est assez vite amené à objecter que si faible P_{occ} soit-elle, on aura toujours par exemple :

$$P_{cheminlongeantlebord} \simeq P_{occ}^{2N} \quad (1)$$

ce qui est rigoureusement non nul. Aussi le seuil de percolation n'a-t-il de sens que pour des réseaux de taille $N \rightarrow +\infty$. A cette condition seulement, par passage à la limite, on retrouve une probabilité nulle pour les chemins triviaux et la notion de seuil de percolation devient pertinente. Il faut alors la présence d'un nombre infini de chemins possibles afin de compenser la probabilité asymptotiquement nulle de chacun. Le seuil de percolation constitue alors intuitivement une sorte de marche d'escalier :

$$\begin{cases} P < P_{seuil} \Rightarrow P_{perc} \\ P > P_{seuil} \Rightarrow P_{perc} \simeq 1 \end{cases}$$

Cette variation brutale est directement reliée au Théorème 0-1 de Kolmogorov. Dans notre cas bien évidemment, impossible de travailler sur un réseau de taille infinie. Toutefois, pour une taille suffisamment grande ($N^2 > 200$), il eût sans doute été possible de réaliser une étude fiable avec des résultats significatifs. Malheureusement, nous n'avons pas eu le temps de construire des réseaux de plus de $N^2 = 25$ billes, et nous n'avons donc pas pu véritablement mener ce travail à bien.

4 Modèles informatiques

Afin d'obtenir des valeurs relatives théoriques d'intensité en sortie de prisme final, on considère un modèle simple dans lequel la lumière, à chaque contact entre deux cylindres ou entre un cylindre et une plaque, peut subir deux phénomènes différents. Le premier, associé à une proportion $\mathbf{1-d}$ de lumière incidente, correspond au trajet logique, non perturbé par la présence du contact : la lumière passe dans le cylindre suivant sans encombre et tourne dans le sens inverse de celui qu'elle avait juste avant. Le second, touchant une proportion \mathbf{d} de la lumière incidente, traduit la diffusion de la lumière au contact : on considère que cette partie est réémise dans toutes les directions par diffusion. Une fraction \mathbf{p} de cette lumière est alors envoyée sous l'angle critique. De façon symétrique, 50% continue alors son chemin dans le sens horaire, et 50% dans le sens anti-horaire. Il est à noter que le prisme de sortie est unique et situé d'un seul côté de la plaque de sortie. Ainsi, la lumière tournant dans le mauvais sens dans le dernier cylindre partira également dans le mauvais sens dans la plaque et ne sera pas collectée par le prisme. Il faut ainsi ne tenir compte dans le calcul de l'intensité de sortie que des chemins aboutissant à la bonne direction de la lumière dans la plaque. Enfin, le nombre moyen de réflexions \mathbf{n} dans les cylindres (donné en première grosse approximation par l'angle d'entrée θ_0) fixe la proportion de lumière conservée dans le cylindre à l'aide du coefficient de réflexion intrinsèque au matériau \mathbf{R} .

Le programme MATLAB ci-dessous calcule l'intensité théorique I_{theo} (normalisée par l'intensité entrante) suivant différentes valeurs de \mathbf{d} et \mathbf{p} en fonction du nombre de cylindres dans le chemin rectiligne envisagé (N_{max}), et détermine la valeur du couple $[\mathbf{d}, \mathbf{p}]$ qui minimise l'écart aux données expérimentales à \mathbf{R} fixé, et ce pour 6 valeurs possibles de θ_0 . Il superpose ensuite I_{theo} et I_{exp} pour les 6 angles d'entrée. Pour clarifier les variables, \mathbf{d} est appelée dans le programme **diffusivite** et \mathbf{p} y est nommée **directivite**. Le programme fait appel à deux sous-fonctions ('Ii' et 'Ip') calculant respectivement l'intensité de sortie pour les nombres impairs de cylindres et celle pour les nombres pairs.

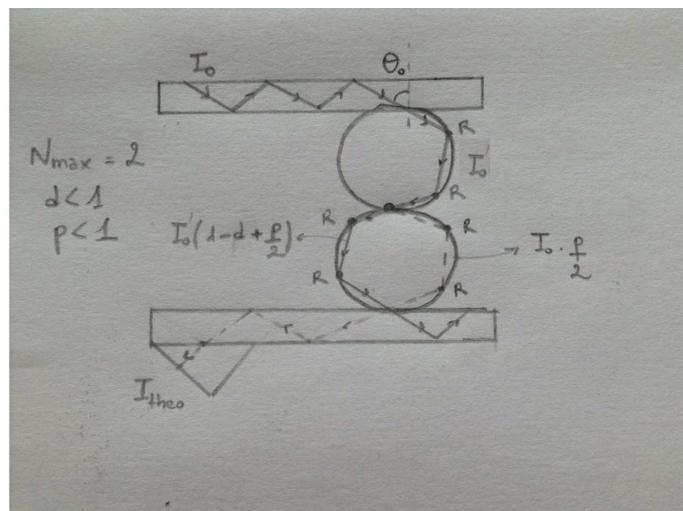


Figure 8: Dessin représentant les paramètres du calcul théorique

Le programme nous retourne les courbes suivantes. Celles-ci mettent en évidence la corrélation plutôt bonne entre le modèle théorique et l'expérience, et mettent en lumière une enveloppe de décroissance de l'intensité suivant le nombre de cylindres, modulée par la parité de ce dernier, ainsi que le modèle le prévoyait.

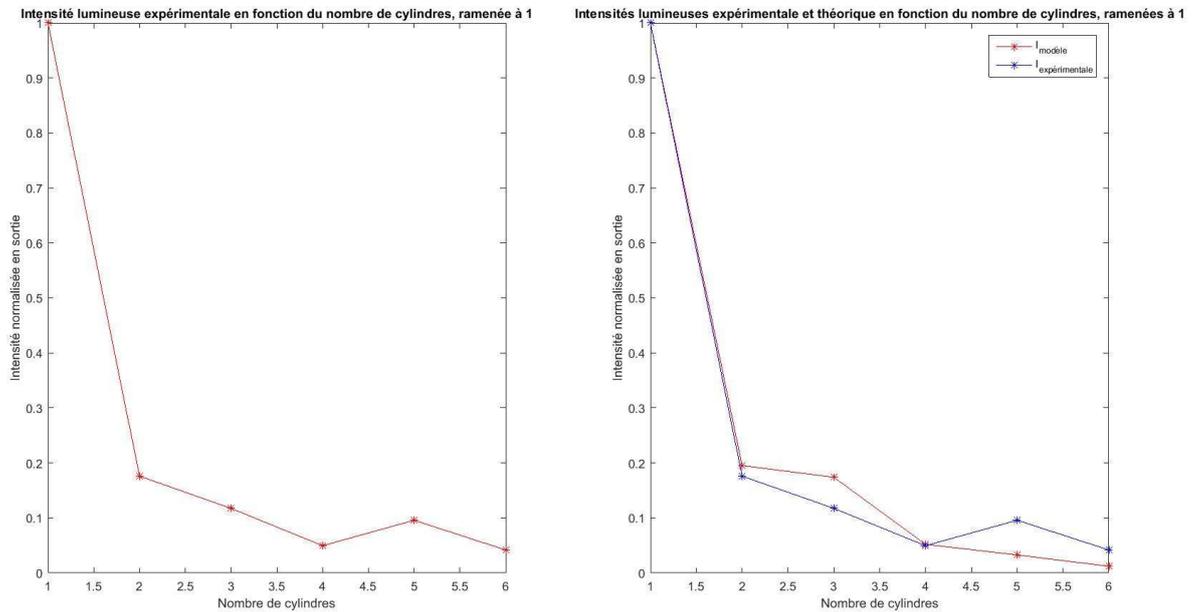


Figure 9: Graphe de l'intensité de la lumière reçue en fonction du nombre de cylindres

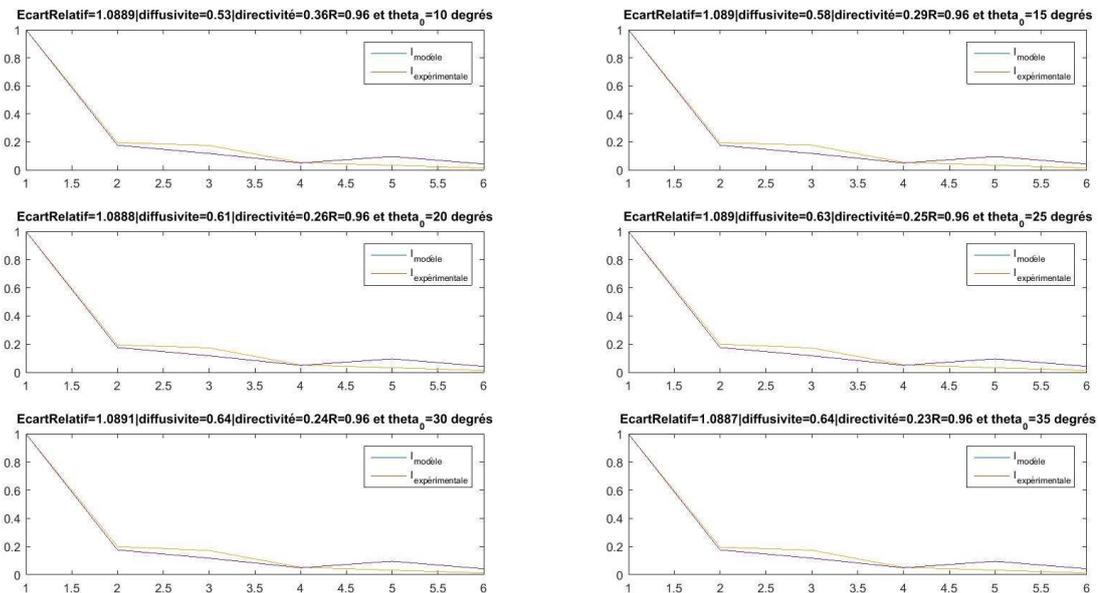


Figure 10: Graphe de l'intensité de la lumière reçue en fonction du nombre de cylindres et des paramètres

4.1 Programme MATLAB principal 'pse.m'

```

clear all
clf
clc
global R;

%theta_0=20;
R=0.96;
Nmax=6;
for v=1:6
    theta_0=5*(1+v);
    nref=90/theta_0;
    figure(84);
%% détermination solution expérimentale

Iexp=zeros(1,Nmax);
Itheo=zeros(1,Nmax);
Inp=zeros(1,floor((Nmax+1)/2));
Ini=zeros(1,floor((Nmax+1)/2));
A=zeros(1,Nmax);

Iexp=[648 114 76 32 62 27]/648;

SigmaDir=zeros(10,3);
for directivite=[0.1:0.1:1]

B=zeros(1,10);
SigmaDiffPrecis=zeros(1,21);
for diffusion=[0.1:0.1:1]

for i=0:floor((Nmax-1)/2)
    Ini(i+1)=Ii(2*i+1,diffusion,directivite,nref);
    if(i>0)
        Ini(i+1)=Ini(i+1)/Ini(1);
    end;
end;
for i=1:floor((Nmax)/2)
    Inp(i)=Ip(2*i,diffusion,directivite,nref);
    Inp(i)=Inp(i)/Ini(1);
end;
for(i=1:Nmax)
    if (i/2==floor(i/2))
        A(i)=Inp(ceil(i/2));
    else
        A(i)=Ini(ceil(i/2));
    end;
end;
end;
A(1)=1;

Itheo=A;
Iecart=(Iexp-Itheo)./Iexp;

```

```

B(round(diffusion*10))=sqrt(Iecart*Iecart');
end;
minErreur=find(B==min(B))/10;

for diffusion=[minErreur-0.1:0.01:minErreur+0.1]
for i=0:floor((Nmax-1)/2)
    Ini(i+1)=Ii(2*i+1,diffusion,directivite,nref);
    if(i>0)
        Ini(i+1)=Ini(i+1)/Ini(1);
    end;
end;
for i=1:floor((Nmax)/2)
    Inp(i)=Ip(2*i,diffusion,directivite,nref);
    Inp(i)=Inp(i)/Ini(1);
end;
for(i=1:Nmax)
    if (i/2==floor(i/2))
        A(i)=Inp(ceil(i/2));
    else
        A(i)=Ini(ceil(i/2));
    end;
end;
A(1)=1;

Itheo=A;
Iecart=(Iexp-Itheo)./Iexp;
SigmaDiffPrecis(round((diffusion-(minErreur-0.1))*100+1))=sqrt(Iecart*Iecart');
if diffusion>=1
SigmaDiffPrecis(round((diffusion-(minErreur-0.1))*100+1))=20000000;
end;
end;

iMinErreur=find(SigmaDiffPrecis==min(SigmaDiffPrecis));
minDiff=minErreur-0.1+0.01*iMinErreur;

SigmaDir(round(directivite*10),1)=SigmaDiffPrecis(iMinErreur);
SigmaDir(round(directivite*10),2)=minDiff;
SigmaDir(round(directivite*10),3)=directivite;
end;

iMinErreurDir=find(SigmaDir==min(SigmaDir(:,1)));
minDiff=SigmaDir(iMinErreurDir,2);
minDir=SigmaDir(iMinErreurDir,3);

for directivite=[minDir-0.1:0.01:minDir+0.1]

for diffusion=[0.1:0.1:1]

for i=0:floor((Nmax-1)/2)

```

```

    Ini(i+1)=Ii(2*i+1,diffusion,directivite,nref);
    if(i>0)
    Ini(i+1)=Ini(i+1)/Ini(1);
    end;
end;
for i=1:floor((Nmax)/2)
    Inp(i)=Ip(2*i,diffusion,directivite,nref);
    Inp(i)=Inp(i)/Ini(1);
end;
for(i=1:Nmax)
    if (i/2==floor(i/2))
        A(i)=Inp(ceil(i/2));
    else
        A(i)=Ini(ceil(i/2));
    end;
end;
A(1)=1;

Itheo=A;
Iecart=(Iexp-Itheo)./Iexp;
B(round(diffusion*10))=sqrt(Iecart*Iecart');
end;
minErreur=find(B==min(B))/10;

for diffusion=[minErreur-0.1:0.01:minErreur+0.1]
for i=0:floor((Nmax-1)/2)
    Ini(i+1)=Ii(2*i+1,diffusion,directivite,nref);
    if(i>0)
    Ini(i+1)=Ini(i+1)/Ini(1);
    end;
end;
for i=1:floor((Nmax)/2)
    Inp(i)=Ip(2*i,diffusion,directivite,nref);
    Inp(i)=Inp(i)/Ini(1);
end;
for(i=1:Nmax)
    if (i/2==floor(i/2))
        A(i)=Inp(ceil(i/2));
    else
        A(i)=Ini(ceil(i/2));
    end;
end;
A(1)=1;

Itheo=A;
Iecart=(Iexp-Itheo)./Iexp;
SigmaDiffPrecis(round((diffusion-(minErreur-0.1))*100+1))=sqrt(Iecart*Iecart');
if directivite>1
    SigmaDiffPrecis(:)=SigmaDiffPrecis(:)*1000000;
end;

```

```

end;
iMinErreur=find(SigmaDiffPrecis==min(SigmaDiffPrecis));
minDiff=minErreur-0.1+0.01*iMinErreur;
SigmaDirPrecis(round((directivite-(minDir-0.1))*100+1),1)=SigmaDiffPrecis(iMinErreur);
SigmaDirPrecis(round((directivite-(minDir-0.1))*100+1),2)=minDiff;
SigmaDirPrecis(round((directivite-(minDir-0.1))*100+1),3)=directivite;
end;
iMin=find(SigmaDirPrecis==min(SigmaDirPrecis(:,1)));
minTotDiff=SigmaDirPrecis(iMin,2);
minTotDir=SigmaDirPrecis(iMin,3);
%% Dessin modèle

diffusion=minTotDiff;
directivite=minTotDir;

Nmax=6;

Itheo=zeros(1,Nmax);
Inp=zeros(1,floor((Nmax+1)/2));
Ini=zeros(1,floor((Nmax+1)/2));
A=zeros(1,Nmax);

for i=0:floor((Nmax-1)/2)
    Ini(i+1)=Ii(2*i+1,diffusion,directivite,nref);
    if(i>0)
        Ini(i+1)=Ini(i+1)/Ini(1);
    end;
end;
for i=1:floor((Nmax)/2)
    Inp(i)=Ip(2*i,diffusion,directivite,nref);
    Inp(i)=Inp(i)/Ini(1);
end;
for(i=1:Nmax)
    if (i/2==floor(i/2))
        A(i)=Inp(ceil(i/2));
    else
        A(i)=Ini(ceil(i/2));
    end;
end;
A(1)=1;

x=1:Nmax;

subplot(3,2,v);
plot(x,A);
hold on;
plot(x,Iexp);
title(['EcartRelatif=',num2str(SigmaDirPrecis(iMin,1)), '| diffusivite=',num2str(minTotDiff), '| d
legend('I_{modèle}', 'I_{expérimentale}');
end

```

4.2 Sous-fonction 'Ii.m'

```
function [I,I1]=Ii(N,d,p,nref)
global R;
alpha=(d*p/2+1-d);
if(N/2==ceil(N/2))
    N=2*N+1;
end
n=(N-1)/2;
for k=0:n+1
    ii(k+1)=nchoosek(2*n+2,2*k)*alpha^(2*k)*(d*p/2)^(2*n+2-2*k);
end;
for k=0:n
    iip(k+1)=nchoosek(2*n+2,2*k+1)*alpha^(2*k)*(d*p/2)^(2*n-2*k);
end;
I=sum(ii);
I=I*((R^N)^nref);
I1=sum(iip);
end
```

4.3 Sous-fonction 'Ip.m'

```
function [I,I1]=Ip(N,d,p,nref)
global R;
alpha=(d*p/2+1-d);
if(N/2~=ceil(N/2))
    N=2*N;
end
n=N/2;
for k=0:n
    ip(k+1)=nchoosek(2*n+1,2*k)*alpha^(2*k)*(d*p/2)^(2*n+1-2*k);
end;
for k=0:n-1
    ipp(k+1)=nchoosek(2*n+1,2*k+1)*alpha^(2*k+1)*(d*p/2)^(2*n-2*k);
end;
I=sum(ip)*((R^N)^nref);
I1=sum(ipp);
end
```